AD-A232 945

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
		FINAL TECH RPT, 15 Sep 89 to 14 Dec 90

4. TITLE AND SUBTITLE
MOTION ANALYSIS AND ITS APPLICATIONS

5. FUNDING NUMBERS
F49620-89-C-0126
62301E
6227 03 DARPA

6. AUTHOR(S)
Ram Nevatia

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Institute for Robotics & Intelligent Systems
University of Southern California
Powell Hall, Room 204
Los Angeles, CA 90089-0273

8. PERFORMING ORGANIZATION REPORT NUMBER
AFOSR-TR- 91 0187

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NM
Bldg 410
Bolling AFB DC 20332-6448

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
F49620-89-C-0126

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release;
distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This research project has addressed motion analysis and its applications with the study of techniques to detect, track, and predict the motion of moving objects from a moving platform. For this project, the major goal was the development of techniques for describing a three-dimensional environment using a sequence of images from a mobile robot. This goal was attached by several separate research projects in motion analysis, which also used genera' image analysis techniques from our other (past or current) research projects. This report will describe the overall direction of our motion analysis research with descriptions of our major recent results.

14. SUBJECT TERMS

91 3 08 025

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

Motion Analysis and Its Applications

Contract F49620-89-C-0126
Final Technical Report
September 15, 1989 to December 14, 1990
DARPA Order 6227
Program Code 8E20
Contractor: University of Southern California
Start Date: 9/15/89
Expiration Date: 12/14/90
Contract Dollars: $401,885
Principal Investigator: Ram Nevatia
(213) 743-5516
Program Manager: Abraham Waksman
(202) 767-5025

R. Nevatia and K. Price (Editors)
Institute for Robotics and Intelligent Systems
School of Engineering
University of Southern California
Los Angeles, CA 90089-0273

January, 1991

Sponsored by Defense Advanced Research Projects Agency DARPA Order No. 6227
Monitored by AFOSR Under Contract No. F49620-89-C-0126

1 Preface

This report describes our research activities on Contract F49620-89-C-0126 for the period September 15, 1989 through December 31, 1990, and is the "Final Technical Report." This Final Technical Report presents an overview of the work of the entire period of the contract. Our basic approach to detecting and tracking motion is to extract and match features, such as lines and regions, from a sequence and to generate motion estimates from these correspondences.

2 Introduction and Summary

This research project has addressed motion analysis and its applications with the study of techniques to detect, track, and predict the motion of moving objects from a moving platform. For this project, the major goal was the development of techniques for describing a three-dimensional environment using a sequence of images from a mobile robot. This goal was attacked by several separate research projects in motion analysis, which also used general image analysis techniques from our other (past or current) research projects. This report will describe the overall direction of our motion analysis research with descriptions of our major recent results. This report represents the developments of the September 15, 1989 to December 14, 1990 period.

Two basic approaches to motion analysis are the short range (optical flow) and long range (feature point) methods. Observations by psychologists, that the relative motion or flow of scene points as projected on the retina can determine the relative depth of objects, led computer vision researchers to the idea of optical flow. Optical flow computations are limited to small motions between views and have proven to be unstable and unreliable in the general (real image) case. These methods are appealing in a mobile robot task for computing the vehicle motion since global techniques tend to reduce errors, but these methods have not been able to overcome their basic computational problems in this case.

The other methods, called feature point or long range techniques, attempt to compute many of the same properties as optical flow methods using far fewer points in each image. These use a small set of corresponding points from the image sequence to compute the three-dimensional motion and structure. Different methods require different numbers of points in various numbers of frames under different assumptions. Generally, a set of equations, which encapsulate the constraints imposed by the assumptions (rigidity, small motions, etc.), are solved to derive the three-dimensional motion parameters. These formulations are sensitive to noise in the input data and produce unstable results, especially when only two views of the scene are used. In order to capture the important constraints imposed by an extended sequence of views, we developed a technique to estimate the motion parameters using five frames for general motion and three frames for translational motion [Sharit86]. We continue to use this underlying motion computation and to develop multi-frame techniques.

Motion analysis using feature point analysis techniques and multiple frames forms the central focus of our work. This approach involves extracting a set of consistent features from a sequence of images, finding the corresponding features in consecutive frames, and finally computing the three-dimensional motion based on the correspondences, which also provides an estimate of the structure of the moving objects or scene. These are often described separately or as sequential operations, but integration into a single system and feedback to earlier processing is a major part of the work.

Our effort includes several separate and related projects including: analysis of closely spaced images (spatio-temporal analysis) using features such as lines, corners, and regions to ex-

tract three-dimensional structure information; matching edge based contours in a sequence of images; integrating several feature detection and matching techniques to derive three-dimensional motion and structure estimates; study of the formulation of the motion estimation problem; detection of moving objects in a scene with a moving observer; and the visual guidance of a mobile robot.

This report presents the results in the major research components of our motion analysis work. The results of this research have also appeared in other conferences and workshops. The work described in the report and the writing of the report represent the efforts of several researchers in our group. The feature matching work was performed by Salit Gazit with Gerard Medioni. The spatio-temporal analysis is by Shou-Ling Peng with Gerard Medioni. The motion system development was done by Yong Kim and Keith Price, the motion estimation work was done by Wolfgang Franzen, and the mobile robot work has been primarily by Jean-Yves Cartoux.

2.1 SPATIO-TEMPORAL ANALYSIS

The goal of our work in spatio-temporal analysis is to generate a dense optic flow map from a motion sequence. Because of the sparseness of 0D features (corners) or 1D features (curves), we feel 2D features (regions) are more likely to produce dense motion estimates.

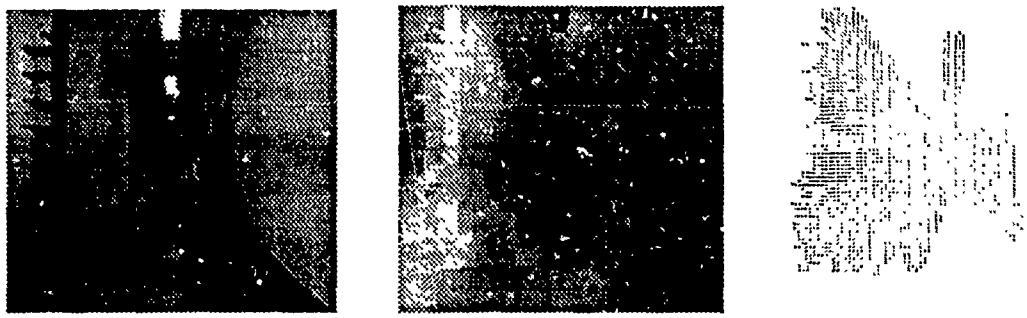
Early work in spatio-temporal analysis includes that of [Bolles87] and depended on knowing the camera motion and restricted this motion to simple translations These techniques use the close spacing to simplify the computation of correspondences between frames -- the corresponding feature is the closest one in the next frame.

The basis for analysis is matching image features along slices cut through the time-image volume of data. Assuming the motion can be approximated by piecewise translational motion along the camera axis, and the focus of expansion (FOE) position is given, the motion direction of each image element corresponding to a stationary object is given. If slices are cut at the FOE along the direction radiating from the FOE, the match disparities are then the magnitude of the image plane velocity, giving a dense optic flow map.

Since the spatio-temporal images are registered by a rectangular coordinate system, cutting radial slices is equivalent to transforming the images into a polar coordinate system. This causes resolution problems: if the slices through the sequence are dense enough so that pixels far from the FOE are sampled by one slice, then pixels close to the FOE are included in many slices.

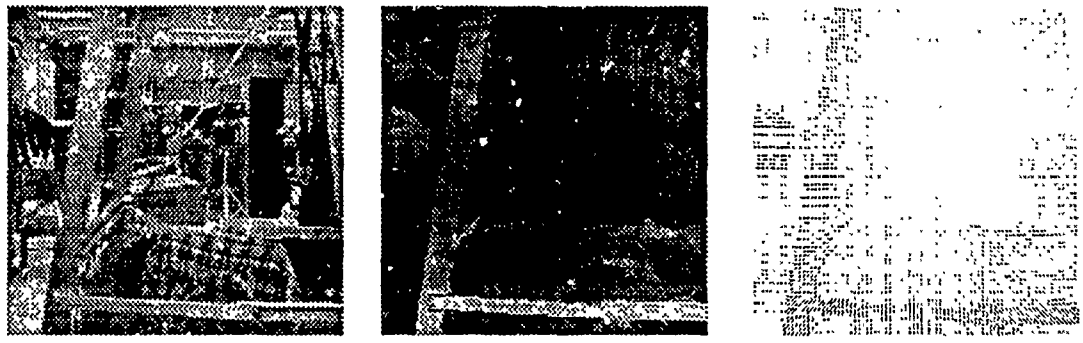
If we examine the slices more carefully, some pairs of paths serve as the non-parallel sides of trapezoidal regions. Each such region corresponds to a collection of chords of a moving object seen in each image in the sequence. If we assume that the velocity changes smoothly between two paths (i.e. between two points on an object), we generate flow values for all pixels in the region by interpolation.

Assuming the motion in the scene is approximated by piecewise translational motion along the axis of the camera, and the focus of expansion (FOE) position is given, the optical flow direc-



First Frame Velocity Needle Diagram

Figure 1. SRI Sequence: Hallway



First Frame Velocity Needle Diagram

Figure 2. SRI Sequence: Zoom

tion of each image element can be determined. If slices are cut at the FOE along the direction radiating from the FOE and image points are matched in the slice, the match disparities are then the magnitude of the velocity. This, when combined with the interpolation, produces a dense optic flow map.

We devised a parallel algorithm to approximate the complete radial slicing, which simplifies this data access problem. We only take slices at each pixel along the four directions: horizontal, vertical, 45°, and -45°. Using the interpolation step mentioned above, each pixel would have at most four estimates of the velocity components along different directions. Using the method presented earlier in [Peng88,Peng89], the normal velocity of the pixel is recovered. With both the motion direction (from the FOE position) and normal velocity (from the slice analysis), we are able to compute the velocity of the pixel. From our experiments, the results from both approached are very similar. Results of using this technique are shown in Figure 1 and 2 showing one frame of a sequence, the computed velocity for the optical flow and the typical optical flow direction diagram.

2.2 FEATURE-BASED MOTION CORRESPONDENCE

Feature matching is a major component of any feature based motion system. We have developed and used several different general feature matching methods in the past. Under this re-

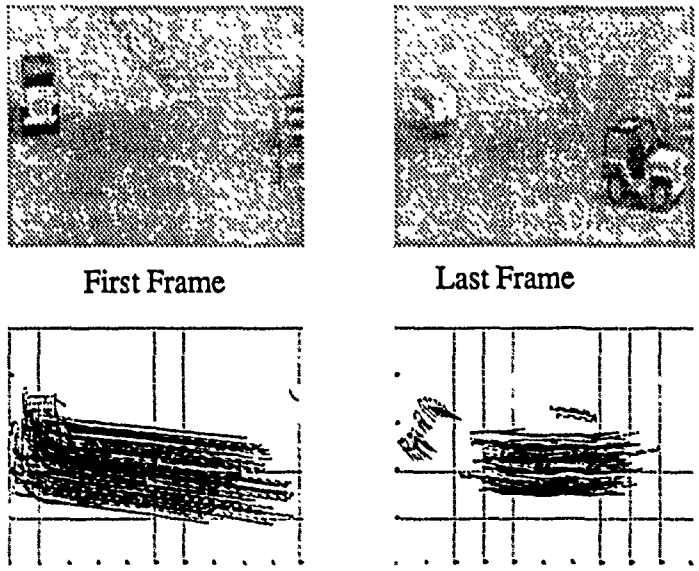
search program we are developing a contour based matching approach that uses large scale matches to guide the finding of detailed edge element matches for images with spacings greater than the spatio-temporal approaches. This approach begins by finding matches for line segment approximations of the edge contours in the images. Then portions of the contour are matched using the approximate matches given by the segment matches to limit the final edge element to edge element matches. The set of pair-wise matches are combined to generate traces of matching edge elements through the entire sequence. Matches through parts of the sequence are also maintained.

This contour-based matching technique is derived from our earlier work [Gazit88,Gazit89]. The major improvements are that multiple frame matches need not extend through the entire sequence of images, thus allowing for occlusion and (re)appearance of points midway through the sequence; and the use of *neighborhood and length* to distinguish between correct and incorrect matches. These changes have resulted in a significant improvement in the quality of the matches.

A brief description of the contour matching method is: A *super-segment is* an object described both as a list of connected edgels and a list of connected line-segments (that approximate the edgel contour). The algorithm tries to match sections of super-segments. Since a single object may correspond to several different super-segments and a single super-segment may include more than one object, the problem is to identify the matching super-segment *sections*. We base our initial matching criterion only on *shape similarity* and p*roximity* (with a maximum allowable disparity). An initial approximation is found by first matching the line-segments and combining matches along each super-segment. Next we compute the section matches themselves. In order to find appropriate matching sections, we break the line-segment approximations used in the previous stage into arbitrary small sections and match them (along the possibly matching section) by maximizing the similarity between the matching sections as well as the length (in points) of the matching sections. The result is a very large set of matches, the great majority of which are spurious. The main thrust of the work is in how to deal with these spurious matches.

Our solution to distinguish between incorrect and correct matches is based on the assumption that correct matches will usually either be l*ong* or will have a*pproving neighbors*, which are neighbor matches representing a similar motion. The neighborhood size should ideally depend on object size, but since this step comes before object segmentation, we instead use a fixed fraction of the image size. Each match is assigned a *approving length* score which is a combination of the total length of supporting neighboring matches and their number; a *non-approving length* score computed from the non-supporting neighbor matches; and a *shape similarity* score. Using these three measures together allows us to detect incorrect matches in most of the cases, since they are short, have little neighborhood support and a lot of neighborhood rejection, as they represent an inconsistent motion. A notable exception to this occurs with straight line contours, which are easy to detect and for which we have a partial solution, and repetitive structures.

We apply this algorithm hierarchically for different scales for better performance [Gazit89]. We also combine pairwise matches into multiple-frame matches by a tracking the matching sections through the sequence. If section P_1 in frame 1 matches section P_2 in frame 2,



First Frame Last Frame

Multiple Matches of the Jeep File Multiple Matches of the Train

Figure 3. Jeep and train image - Multiple Matches

Q_2 in frame 2 matches Q_3 in frame 3 and P_2 overlap Q_2, we can compute a new match (P_2,R_3) corresponding to the overlapping part. This is applied to all frames in the image sequence. The resulting section matches can be used for 3D motion estimation or motion segmentation.

As an example, a sequence consisting of 10 250 × 512 frames of a toy jeep and train is given in figure 1. We only show the last two frames and the resulting multiple matches. Because the motion of the objects overlap, and also to allow for better visibility, we manually removed the background matches and separated the matches corresponding to the train and those corresponding to the jeep.

In this scene the camera is stationary, but both the jeep and the train are moving. This is a difficult scene as the motion is very large (disparity ranges from 0 to 150 pixels) and the scene contains occlusion.

2.3 DETECTING MOVING OBJECTS FROM A MOVING PLATFORM

Detecting moving objects from a moving platform is a difficult problem, because the observer motion causes stationary objects to appear to move. Thus, we must separate genuine motion from the apparent motion of the stationary environment. We have developed a system that successfully detects moving objects within a sequence of real images taken from an observation vehicle traveling along a road. Unlike other systems, ours does not require densely-sampled imagery, meaning that objects can move many pixels per frame with no detrimental effects. Nor does the system rely on critical parameter settings. It is computationally efficient, and highly suited to parallel implementation. It requires no object matching or recognition, and can thus detect moving objects that are partially occluded or that are camouflaged.

When an observer moves in a straight line, toward a distant point in space, stationary objects in the environment appear to move along paths radiating from that point. The point from which the paths radiate is called the focus of expansion (FOE). We assume that the FOE, and camera orientation, are relatively stable between successive images (i.e., the observer must not sharply turn or tilt between images).

To simplify the problem, we first perform a Complex Logarithmic Mapping (CLM) as suggested by [Cavanaugh78,Weiman79,Jain84]. This converts the problem from one of detecting a complex motion along both the X and Y axes, to one of detecting motion along an angular axis, with stationary objects moving along the other axis.

To detect the angular motion in CLM space, we have developed a novel "moving-edge detector," which operates on successive images and produces a map containing all pixels on the edge of regions that are moving relative to the stationary background. This map is then thresholded to produce detected movement. Preliminary results indicate that, once the threshold is raised high enough to eliminate false alarms, it can be increased by a factor of five and still properly detect moving objects.

The resulting detected movement is then transformed back into the rectangular reference frame, and overlaid upon the original image to highlight the detected objects. The results are presented in more detail in [Frazier90]. This technique depends heavily on the correct computation of the FOE and very loosely on the movement threshold value.

2.4 MOTION ESTIMATION

We have developed a solution for the multiframe structure from motion problem using feature matches. This work assumes a central projection pinhole camera with no smoothness assumptions imposed concerning object surfaces. The use of multiple (as opposed to two) frames is desirable for several reasons:

- to increase the robustness of the solution,
- to allow recovery of structure/motion with fewer features being tracked, and
- to allow estimation of "higher order derivatives" of the motion.

In this work, we have developed and implemented two algorithms to solve the SFM problem that make different assumptions concerning smoothness and type of motion. The first is a closed form algorithm that models the relative motion between the camera and the object or environment as a uniform 3D acceleration. The second is an iterative algorithm that can recover arbitrary rigid transformations between frames. The closed form algorithm is currently being used to generate initial guesses for the iterative algorithm when the rotation is known to be small.

The two algorithms share some characteristics: Both assume that features are matched through at least three frames. The image plane position of each feature is modeled as having a bivariate Gaussian error distribution, with the error coefficients provided as input. Although the algorithms are developed using point features, they can process both point and line features. A given

feature is not required to be visible in every frame, so the algorithms can process features that become (un)occluded during the image sequence. As output, both algorithms generate the 3D location of each feature in each frame, along with the motion parameters.

The closed form algorithm models the motion as a uniform 3D acceleration. It minimizes a norm that is closely related to the maximum-likelihood image plane error norm subject to the constraint that the mean interframe displacement must equal one. Under this formulation, the 3D point positions are linear functions of the motion parameters, and the motion parameters can be determined by solving a small eigenvalue problem. The computational complexity of the algorithm is linear in the number of features being tracked times the number of frames.

The iterative algorithm solves the SFM problem as an unconstrained minimization problem. The function to be minimized consists of 3 (classes of) terms:

1. the image plane error or a more or less convex approximation to it,
2. terms which bias the motion to be chronogeneous or some subclass of chronogeneous motion, and
3. a term which imposes a specific scale on the solution.

Minor changes in the form of a term may dramatically alter the convergence properties of the algorithm. The algorithm is currently very slow because analytic derivatives have not been programmed, and a quasi-Newton method with a finite difference gradient is being used to do the optimization.

2.5 INTEGRATED SYSTEM FOR MOTION

We have developed an integrated system for testing each of the subsystems of the motion analysis system (segmentation, feature extraction and matching, motion estimation, motion feedback to matching and coordination). The results of each subsystem is saved in a single data structure and the coordination module controls exchange of information between subsystems. The integrated system is now being used to generate a rough description of three-dimensional structure of the environment, using region-based matches refined by corner matches over multiple frames. This work is described in more detail in these proceedings in [Kim90].

Feature matching is done in a coarse-to-fine manner to reduce search space and enhance stability. Corner-based matching for a region is guided by the motion computed for the centers of mass of the matched regions and by the constraint that matching corners are on the same region. This allows large disparities between images and different motions for each of the regions. Corner-based matching is performed both in the forward and reverse directions to decrease errors in matching.

We have developed a translation dominant motion analysis system as an additional feature of the general motion analysis system. The basic assumptions are that each object in the scene is undergoing a translation dominant motion and that an object may (or may not) be in coherent motion with some of the others. An approximate FOE (focus of expansion) using a LMSE (least mean

square error) estimation and motion parameters are estimated for each region and then depth is computed for the corners of the region. Each computed result is associated with a reliability factor, which is a measure of the closeness to the computed motion to a translational motion. Regions with a high reliability are given high priorities in the analysis and their results act as a guide in the analysis of the less reliable regions by giving some constraints to the motion parameters.

This motion analysis system was tested for two real image sequences. A camera is moving straight along a hallway in one of them, and in the other sequence, a car is moving from the right side of the image to the other end. With a reasonable amount of noise, we could obtain an approximate environmental depth map for most of the important regions in the scene. Depth maps with region-corner matches are shown in [Kim90].

Experiments show some weak points for this system. First, the use of the FOE analysis for general motion (translation + rotation) is sensitive to noise and thus the computed motion parameters are numerically unstable. In the case of translation dominant motion, an accurate estimation of FOE is essential for reliable results. Second, information of depth is lost along a smooth boundary even when it forms a great part of a region since fine structure is determined by corner matches.

We continue to add more features to our integrated system. Primarily, we plan to add more feedback links within the system so that an erroneous matches at an early stage is detected and corrected by results of later stages. This way, motion analysis is done as a part of a cooperative process rather than an isolated stage of a sequential process.

2.6 MOBILE PLATFORM

We acquired a Denning mobile robot for both indoor and outdoor experimentation. The initial phase of experimentation dealt with basic control and navigation issues but the goals include visual feature navigation and a platform for testing our other motion algorithms. We do not intend to concentrate on real-time (high speed) control, which would only be possible with additional special purpose computers, but to develop high-performance analysis algorithms. This initial effort has produced:

- an obstacle avoidance routine using the range data provided by the 24 ultrasonic sensors of the robot, and
- a simple planner allowing the robot to navigate indoors.

An obstacle in front of or on the sides of the robot is detected by checking the ultrasonic sensors in near the direction of motion. If there is an obstacle, the robot turns toward the direction of the first sensor where the path is clear. This is intended as a low-level survival process rather than a major navigational tool.

The map of the robot world is represented by a hierarchical data structure that includes buildings which are defined by a set of floors. Each floor has hallways, a set of rooms and a set of walls. Each wall may include doors.

In the first phase, the robot is assigned to navigate in the hallway of a floor. The ideal trajectory is the mid-line between the two walls of the hallway. The planner first computes a list of the axis of symmetry of each hallway path. Each axis is limited to the common part of the pair of walls, must be inside the external polygon of the hallway, but not inside any of its internal polygons. A merging step produces the axes of the corridors.

From the extremes and the intersections points of these axes, a graph of trajectory control points is constructed. The path of the robot, shown by thick black circles and lines on the figure, from its current location toward a goal door is then computed from the graph representation. Finally, the list of path control points is given to the navigation routine that orients the robot toward the next path control point, unless the robot is bypassing an obstacle.

3 Bibliography

[Bolles87] Robert C. Bolles, Harlyn Baker, and David H. Marimont, "Epipolar-plane Image Analysis: An Approach to Determining Structure from Motion," *International Journal of Computer Vision, Vol. 1,* No. 1, pp. 7-55, 1987.

[Cavanaugh78] P. Cavanaugh, "Size and Position Invariance in the Visual System," *Perception, Vol. 7,* August 1978, pp. 167-177.

[Frazier90] J. Frazier and R. Nevatia, "Detecting Moving Objects from a Moving Platform," in *Proc. DARPA Image Understanding Workshop,* Pittsburgh, PA, September, 1990, pp. 348-355

[Gazit88] S. L. Gazit and G. Medioni, "Contour Correspondence in Dynamic Imagery," in *Proc. DARPA Image Understanding Workshop,* Boston, MA, April, 1988, pp. 423-432.

[Gazit89] S. L. Gazit and G. Medioni, "Multi-Scale Contour Matching in a Motion Sequence," in *Proc. DARPA Image Understanding Workshop,* Palo Alto, CA, May, 1989, pp. 934-945.

[Jain84] R. Jain, "Segmentation of Frame Sequences Obtained by a Moving Observer," *IEEE Trans. on PAMI, Vol. 6,* 1984, pp. 624-629.

[Kim90] Y. C. Kim and K. Price, "Multiple Frame Analysis of Translation dominate Motion," in *Proc. DARPA Image Understanding Workshop,* Pittsburgh, PA, September, 1990, pp. 339-347.

[Peng88] S. L. Peng and G. Medioni, "Spatio-Temporal Analysis of Velocity Estimation of Contours in an Image Sequence with Occlusion, Proc. ICPR, Rome, Italy, November 1988, pp. 236-241.

[Peng89] S. L. Peng and G. Medioni, "Interpretation of Image Sequences by Spatio-Temporal Analysis," in Proceedings of the IEEE Workshop on Visual Motion, Irvine, California, March 1989, pp. 236-241.

[Shariat90] H. Shariat and K. Price, "Motion Estimation Using More Than Two Frames," *IEEE Trans. PAMI, Vol 12,* No. 5, May1990, pp. 417-434.

[Weiman79] C. Weiman and G. Chaikin, "Logarithmic Spiral Grids for Image Processing and Display," *Computer Graphics and Image Processing, Vol. 11,* No. 3, November 1979, pp.197-226.